# Content Management System (CMS) Project

## Team Members:

Section 5

2320030416 CHITTA ANUBHAV JOSHI

2320030253 KOTA SANJAYKUMAR

2320030440 KAKI LAKSHMI SAKETH

2320090050 N V V S S JAYAKANTH KAMISETTI

2320090054 MANGINA JAGADISH VENKATA SAI

## Problem Statement: Content Management with Time-based Access Control

The ContentManager project is designed to provide a modern solution for managing digital content with a focus on security and flexibility.  
 It incorporates time-based access control mechanisms through key validity periods, ensuring that content is accessible only for predefined  
 time windows. This functionality is particularly beneficial for scenarios where content needs to be restricted based on time, such as   
 subscription-based services or promotional events.

## Features:

1. Modularity:  
 - The system follows a modular architecture, separating concerns into different modules such as APIs, shared utilities, and a frontend.  
 - This design enables easy maintainability and scalability.  
   
 2. API-driven Architecture:  
 - The project integrates a RESTful API that facilitates seamless communication between the frontend and backend, ensuring efficient   
 content management operations.  
   
 3. Frontend with Vue.js and TypeScript:  
 - The frontend is built using modern technologies such as Vue.js and TypeScript, providing a responsive and interactive user experience.

4. Key-based Access Control:  
 - The system allows administrators to define time-bound access to content using keys, enhancing security for sensitive information.

## Development Process:

1. Backend Development:  
 - Java and Gradle are used for backend development, ensuring robust and scalable performance.  
   
 2. Frontend Development:  
 - The UI is developed using Vue.js and TypeScript to provide a smooth user experience.  
   
 3. API Development:  
 - RESTful APIs are created to handle content retrieval, updates, and management.  
   
 4. Security Enhancements:  
 - Implemented key-based access controls to restrict access based on time.

## Expected Outcome:

The ContentManager project is expected to deliver a flexible and secure content management system that offers dynamic key-based   
 access control. The system aims to simplify content management for businesses, ensuring that sensitive information is only   
 accessible during specified time periods. This capability can be beneficial for various industries that require controlled   
 content distribution, contributing to a more secure and efficient content management infrastructure.

**Software Requirements:**

1. **Operating System**:
   * Windows, Linux, or macOS (any modern version).
2. **Development Tools**:
   * **Java Development Kit (JDK)**: Version 8 or higher.
   * **Gradle**: For build automation.
   * **Maven**: (if used) for managing project dependencies.
   * **Node.js**: For frontend development using Vue.js and TypeScript.
3. **Frontend Technologies**:
   * **Vue.js**: JavaScript framework for the frontend.
   * **TypeScript**: For enhanced type checking and scalability of frontend code.
4. **Backend Technologies**:
   * **Spring Boot (Optional)**: To create and deploy RESTful APIs (if applicable).
   * **MySQL/PostgreSQL**: For the content database.
   * **Redis**: (Optional) For caching if needed for performance optimization.
5. **Web Server**:
   * **Apache Tomcat** or any other Java-based server to deploy the application.
6. **Version Control**:
   * **Git**: For version control and collaboration.

**Hardware Requirements:**

1. **Processor**:
   * Minimum: Dual-core processor (2.5 GHz or faster).
   * Recommended: Quad-core processor for smoother development and testing.
2. **Memory (RAM)**:
   * Minimum: 4 GB RAM.
   * Recommended: 8 GB or more for running the development environment, database, and web server concurrently.
3. **Storage**:
   * Minimum: 10 GB of available disk space for project files, dependencies, and development tools.
   * Recommended: 20 GB or more for storing additional libraries, databases, and backups.
4. **Display**:
   * Minimum: 1280x720 resolution.
   * Recommended: 1920x1080 resolution for better workspace management while developing.
5. **Network**:
   * Reliable internet connection for downloading dependencies, version control with Git, and online resources.